FUNCTION	COMMAND		RESPONSE	TROUBLE REPORT
Terminal Control Commands				
Fignion designated user and start work session.  End a work session.  End work session and hold dial-up connection.  End work session and store active workspace.	)123846 )OFF )OFF HOLD )CONTINUE	[:PASSWORD] [:NEWPASSWORD] [:NEWPASSWORD] [:NEWPASSWORD]	Port, time, date, user; system; time, date Port, time, date, user code; time used Port, time, date, user code; time used Time, date, Port, time, date, user code; time used	1 2 3 4 5 7 8 16 16 6 10
End work session, store active workspace, and hold dial-up connection.	) CONTINUE	HOLD [:NEWPASSWORD]	Time, date, user code; time used Port, time, date, user code; time used	6 16
Communication Commands				
Send message to designated port.	) MSGN	PORT (TEXT)	SENT	15 16
Send message to designated port and lock keyboard for reply.	)MSG	PORT (TEXT)	SENT	15 16
Send message to APL Operator.	)OPRN	(TEXT)	SENT	15 16
Send message to APL Operator and lock keyboard for reply.	)OPR	(TEXT)	SENT	15 16

Activate a clear workspace.	) CLEAR			CLEAR WS	Time, Date	16					
Replace active workspace with a copy of a stored WS.	) LOAD	WSID				7	8	16			
Copy an individual function or variable from a stored WS.	) COPY	WSID	NAME	SAVED,	Time, Date	6	7	8	9	10	1
Copy all functions and variables from a stored workspace.	) COPY	WSID		SAVED,	Time, Date	6	7	8	10	16	
Copy an individual function or variable from a stored WS protecting the active workspace.	) PCOPY	WSID	NAME	SAVED,	Time, Date	6	7	8	9	10	1
Copy all functions and variables from a stored workspace protecting the active workspace.	)PCOPY	WSID		SAVED,	Time, Date	6	7	8	10	16	,
Gather functions and variables into a group; first name is name of group.	) GROUP	NAME(S)		NONE		10	11	16	ò		
Erase functions and variables.	) ERASE	NAME(S)		[NOT ERASED.	List of names ]	1.0	16				
Set index origin.	)ORIGIN	0 or 1		WAS.	Former origin	16					
Set maximum number of significant digits for output.	)DIGITS	1 to 16		WAS,	Former maximum	16					
Set size of symbol table in clear workspace.	)SYMBOLS	Positive integer	(min. 26)	WAS.	Former size	16					
Set maximum width for an output line.	) WIDTH	30 to 130		WAS.	Former width	16					
Change Workspace name.	) WSID	Name		WAS,	Former WSID	16					
Re-store a copy of the active workspace.	) SAVE				Time, date, WSID		12	1.3	14	1	6
Store a copy of the active workspace.	) SA VE	WSID		WAS,	Time, date		12				-
Erase a stored workspace from a library.	) DROP	WSID		WAS,	Time, date	_	14			1	

Information Requests				
List names of defined functions.	)FNS		Function names	16
List names of global variables.	) VARS		Variable names	16
List names of groups.	) GRPS		Group names	16
List membership of designated group.	) GRP	NAME	Function names, variable names	16
List halted functions (state indicator).	)SI		Sequence of halted functions	16
List halted functions and associated local variables.	)SIV		Sequence of halted function with names of local variables	16
Give identification of Active workspace.	)WSID		WSID	16
List name of workspaces in designated library (either user's library or a public library).	) LIB	[NUMBER]	Names of stored workspaces	14 16
List ports in use and codes of connected users.	) PORTS		Port numbers and associated user codes	16
List port numbers associated with designated user code.	) PORT	CODE	Port numbers	16

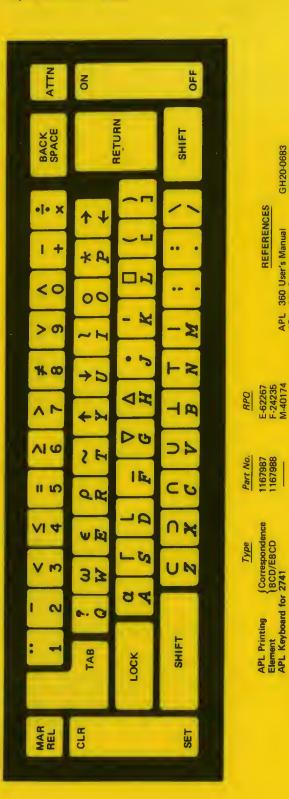
## TABLE 5

2 INCORR 3 ALREAD 4 NUMBER 5 NUMBER	( SIGNED-ON 9 IN USE 1	0 W S		13 14 15	NOT SAVED, QUOTA USED UP NOT SAVED, THIS WS IS IMPROPER LIBRARY REFERENCE MESSAGE LOST INCORRECT COMMAND
-------------------------------------	------------------------	-------	--	----------------	--



APL\360 is a conversational time-sharing system based on a mathematical programming language first defined by Kenneth E. Iverson. The language is concise and has a simple syntax. It has a large set of primitive operations which work directly on arrays. The implementation provides a simple immediate-execution mode and a convenient program definition facility. It has fast response, and uses succinct diagnostic messages. It provides the ability to save work between sessions, to create programming packages, and to exchange programs and data between users. Uses of the system include mathematical and statistical calculation, symbol manipulation, and general data processing. It has been used extensively in computer-aided Instruction, and in the design of hardware and software.





#### APL REFERENCE DATA

	Scalar Dyadi	c Functions				Scalar	Monac	lic Fu	nctions
X + Y	X plus Y			+ )	, ,	Y			
X - Y	The second second			- 3		Y - C			
$X \times Y$	X times Y			×J			Y (1,	0 1	)
X ÷ Y	X divided by	Y		+ ]			cal of		•
	X to the Y-th			* 5			e Y-th		r
XIY	maximum of	X and Y		F 3		eiling		1	
XLY	minimum of	X and Y				loor o		) (Se	e Table 3)
$X \mid Y$	X-residue of	Y (see Table 4)		1.2	۲ i	magnit	ude of	Υ	
	base-X logari			⊕ ]	<i>y</i> 1	natural	logari	thm o	f Y
X ! Y		fficient; for integ		- 11	? 1	actori	al Y; C	iamma	1
		umber of combin	ations		- 1	functio	on of Y	+1	
		aken X at a time							
Хох		hyperbolic funct		0.	Y	π time	es Y		
		erses (Y is in rac	dians)	?				eger f	rom the
	(see Table 1)		_			vector	τY		
	X less than Y			~	Y 1	not Y			
	X less than o								
X = Y			resul	it is	1 i	f the	relatio	n hol	lds,
X≥Y		n or equal to Y	0 if	it d	oes	not			
X > Y									
	X not equal t	(O Y	<u>_</u>	X	У	I Y A Y	XVY	Y av V	YWV
	X or Y				÷	_			
X~Y		nd Y (X nand Y)	<b>}</b>	0	0	0	0	1	1
XWY				0	1	0	1	1	0
л . т	Heither X Ho	1	_	1	0	0	1	1	0
				1	1	1	1	0	0
						Ta	ble 2		
OOY		40Y (1+Y+2)*	.5						
1 o Y		50 Y sinh Y			Y		ГУ	L.	Y
20Y	cos Y	60Y cosh Y		-	_			_	
3 <b>0</b> Y	tan Y	70Y tanh Y			3	.14	4		3
					_3	.14	_3		4
		re given by neg							
values	of X, i.e.	10Y ≡ arcsin \	<i>7</i> .			Ti	able 3		
	Table	1						/ I V	
			_				1	(   Y	
					(≠0			$-X \times L$	<b>Y</b> ÷ X
				)	(=0		Y Y		
						T	able 4		

	XIY
<i>X</i> ≠0	Y-X×L Y ÷ X
X=0	Y
	•

#### Special Symbols

- ( ) Parentheses. Expressions may be of any complexity and are executed from right to left except as indicated by parentheses.
- Branch to X, where X is a scalar or vector. If X is an empty vector, go to. the next line in sequence. If X is not in the range of statement numbers in the function, leave the function.
- Terminate execution of a suspended function,
- □+X Print the value of X. The value of any expression or variable is also printed if no assignment is made.
- X+□ Request input, Value of □ is the resulting value after expression entered is evaluated
- X+1 Request input. Value of 1 is entire input text as literal characters, up to but not including carrier return,

### 'XYZ' The literal characters XYZ,

Underline: Allows increased set of alphabetic characters, i.e., A and A are both distinct characters.

# Printed in U.S.A. S210-0007-0

#### **Mixed Functions**

XρY	Reshape Y to have dimension X
o <b>Y</b>	Dimension of Y
X[Y]	The elements of X at locations Y
$X_1Y$	First location of Y within vector X  The first Y consecutive integers from Origin (0 or 1 as
ı Y	set by set origin command)
$X \in Y$	Each element of X \in Y is 1 or 0 if the corresponding
A ∈ 1	element of X is or is not some element of Y
X T Y	Representation of Y in number system X
KT K Z I Z	Value of the representation Y in number system X
K?Y	X integers selected randomly without repetition from tY
(ΦΥ	Rotation by X along the last dimension of Y
φ[Z]Y	Rotation by X along the Zth dimension of Y
(e)	Rotation by X along the first dimension of Y
Y .	Reversal along the last dimension of Y
[Z]Y	Reversal along the Zth dimension of Y
Y	Reversal along the first dimension of Y
KØ A	Transpose by X of the coordinates of Y
PΥ	Ordinary transpose of Y
Y Y	Y catenated to X
Y	Ravel of Y (make Y a vector)
<b>∴</b> ↑ <b>Y</b>	If X positive take first X elements of Y
	If X negative take last  X elements of Y
<b>+</b> Y	If X positive leave first X elements of Y
	If X negative leave last   X elements of Y
(+Y	X specified by Y
X	The indices of values of the vector X in sorted ascending
, x ·	order The indices of values of the vector X in sorted descending
"	order
	Null
	See Program Definition Section
1	Comment
	In the entries below o stands for "any scalar dyadic
	operator"
	Generalized Reduction
	i.e., insert the symbol o between each pair of elements
	of Y
/Y	The o reduction along the last dimension of Y
/[Z]Y	The or reduction along the last dimension of Y
/ L Z J I	The ereduction along the first dimension of Y
/ 1	The Green along the first differision of 1
	Compression and Expansion
(/Y	
:/1 :/[Z]Y	X (logical) compressing along the last dimension of Y
7 L Z J I	X (logical) compressing along the Zth dimension of Y
Y	X (logical) compressing along the first dimension of Y X (logical) expanding along the last dimension of Y
$\langle [z]y$	X (logical) expanding along the Zth dimension of Y
\Y	X (logical) expanding along the first dimension of Y
,*	77 (10 global) expanding along the first differision of 1
	Generalized Matrix Operations
+ . × Y	Ordinary matrix product of X and Y
o, oy	Generalized inner product of X and Y
0 . OY	Generalized outer product of X and Y
	2 2 3 product of Atlanta
II scalar	functions are extended to operate element-by-element on
mensione	ed operands; i.e., vectors, matrixes, and higher-dimensional

# sion of the other argument.

A scalar or one-component vector may be used as one argument of a

scalar dyadic function and will be extended to conform to the dimen-

\* \* \* \$ \$ I ! \$ \$ \$ \$ \$ \$ A \$ \$ \$

Overstruck Symbols

#### **Definition Mode**

A V (called 'Del') preceding a function name declares a change from execution mode to function definition mode. In definition mode, no execution of commands occurs, and no errors other than character errors, editing errors, and label errors are reported. Instead, each command is stored as part of the definition. A V terminates function defini-

[1]	V <i>T+Q HYP P T</i> +(( <i>Q</i> *2)+ <i>P</i> *2)*0.5∇	dyadic function with explic result
L 1 3	3 HYP 4	result
5	3 1111 4	
	22 - 442 V	n e la la
[1]	∇Z+AVG Y Z+(+/Y)+pY∇	monadic function with expl result
613	AVG 2 4 9 3	result
4.5	A70 2 4 9 3	
643	$\nabla Z + RAN$	niladic function with explic
[1]	Z <b>+?</b> 5p25∇ RAN	result
4 10	3 12 14 6	
· ·	12 14 6	
	∇A HYP B	the second second
[1]	T + ((A * 2) + B * 2) * 0.5V	dyadic function without explicit result
	3 HYP 4	explicit result
	T	
5		
	$\nabla AVG$ N; A	monadic function without
[1]	$A \leftarrow (+/N) + pN$	explicit result; a local varial
[2]	'AVERAGE IS: ';A∇	is defined in the header
	AVG 2 4 9 3	
AVER	AGE IS: 4.5	
	$\forall DRILL$	niladic function without
[1]	'DO THESE PROBLEMS'	explicit result
	11	
[3]	1 1 7	
	DRILL	
DO TI	HESE PROBLEMS	
		and the second s

A function body may be displayed in definition mode:

- [N] Display line N: if N is omitted, display entire function, Mode may be entered, function F displayed, and mode left, by typing  $\nabla F[\Box] \nabla$ .
- $[\square N]$  Display from line N to the end of the function.
- [N[P] Display line N and position the printing element under position P (for editing line N).

#### Trace and Stop Controls

- SAPROG+N Halts execution of PROG immediately before line N is
- TAPROG+N Displays result of calculation performed in line N of func-

N may be a vector. Trace and Stop controls may be removed by N+0 or 10.

### IBM

International Business Machines Corporation Data Processing Division 112 East Post Road, White Plains, New York 10601 (USA only)

IBM World Trade Corporation 821 United Nations Plaza, New York, New York 10017 (International)